Detect & **A**vo**id** **A**lerting **L**ogic for **U**nmanned **S**ystems (DAIDALUS)*

César Muñoz Anthony Narkawicz George Hagen
Jason Upchurch Aaron Dutle María Consiglio

NASA Langley Research Center

The final report of the FAA-sponsored Sense and Avoid (SAA) Workshop for Unmanned Aircraft System (UAS) defines *sense and avoid* as "the capability of a UAS to remain well clear from and avoid collisions with other airborne traffic."[1] The need to provide such a capability has motivated the development of self-separation and alerting algorithms intended to provide remote pilots appropriate situational awareness of proximity to other aircraft in the airspace. This paper presents DAIDALUS (Detect & Avoid Alerting Logic for Unmanned Systems), a reference implementation of a *detect and avoid* (DAA) concept for the integration of civil UAS into the airspace that follows the guidelines provided by the workshop.

At the core of the DAA concept implemented in DAIDALUS, there is a mathematical definition of the well-clear concept. Two aircraft are considered to be *well clear* of each other if appropriate distance and time variables determined by the relative aircraft states remain outside a set of predefined threshold values. These distance and time variables are closely related to variables used in the Resolution Advisory (RA) logic of the Traffic Alert and Collision Avoidance System Version II (TCAS II). A particular set of threshold values determines a corresponding volume around the aircraft, called the violation volume. For some choice of threshold values, the TCAS II RA volume is strictly contained within the well-clear violation volume. That is, aircraft are declared to be in well-clear violation before an RA is issued. The well-clear definition also satisfies the property of *symmetry*, i.e., in a pair-wise scenario, both aircraft make the same determination of being well-clear or not, and the property of *local-convexity*, i.e., in a non-maneuvering pair-wise scenario, there is at most one time interval in which the aircraft are not well clear.

DAIDALUS includes algorithms for determining the current well-clear status between two aircraft and for predicting a well-clear violation within a lookahead time, assuming non-maneuvering trajectories. In the case of a predicted well-clear violation, DAIDALUS also provides an algorithm that computes the

*The acronym DAIDALUS is a reference to the craftsman Daedalus of Greek mythology.
[1]The terms *sense and avoid* and *detect and avoid* are both used in UAS literature.

time interval of well-clear violation. Furthermore, DAIDALUS implements algorithms for computing prevention bands, assuming a simple kinematic trajectory model. Prevention bands are ranges of track, ground speed, and vertical speed maneuvers that are predicted to be in well-clear violation within a given lookahead time. These bands provide awareness information to remote pilots and assist them in avoiding certain areas in the airspace. When aircraft are not well clear, or when a well-clear violation is unavoidable, the DAIDALUS prevention bands algorithms compute well-clear recovery bands. Recovery bands are ranges of horizontal and vertical maneuvers that assist pilots in regaining well-clear status within the minimum possible time. Recovery bands are designed so that they do not conflict with resolution advisory maneuvers generated by systems such as TCAS II.

Finally, DAIDALUS implements two alternative alerting schemas. One schema is based on the prediction of well-clear violations for different sets of increasingly conservative threshold values. The second schema is based on the types of bands, which can be either preventive or corrective, computed for a single set of threshold values. A band is preventive if it does not include the current trajectory. Otherwise, it is corrective. Recovery bands, by definition, are always corrective. In general, both schemas yield alert levels that increase in severity as a potential pair-wise conflict scenario evolves.

All algorithms included in DAIDALUS are implemented in C++ and Java and the code is available under NASA's Open Source Agreement. The implementations are modular and highly configurable via the DAIDALUS application programming interface (API). The DAIDALUS algorithms have been formally specified in a mathematical notation and verified for correctness in a theorem prover. The software implementations have been validated against the formal models using randomly generated unit test cases. DAIDALUS is under consideration for inclusion in the appendix of RTCA Special Committee 228 Minimum Operational Performance Standards (MOPS) for Unmanned Aircraft Systems.